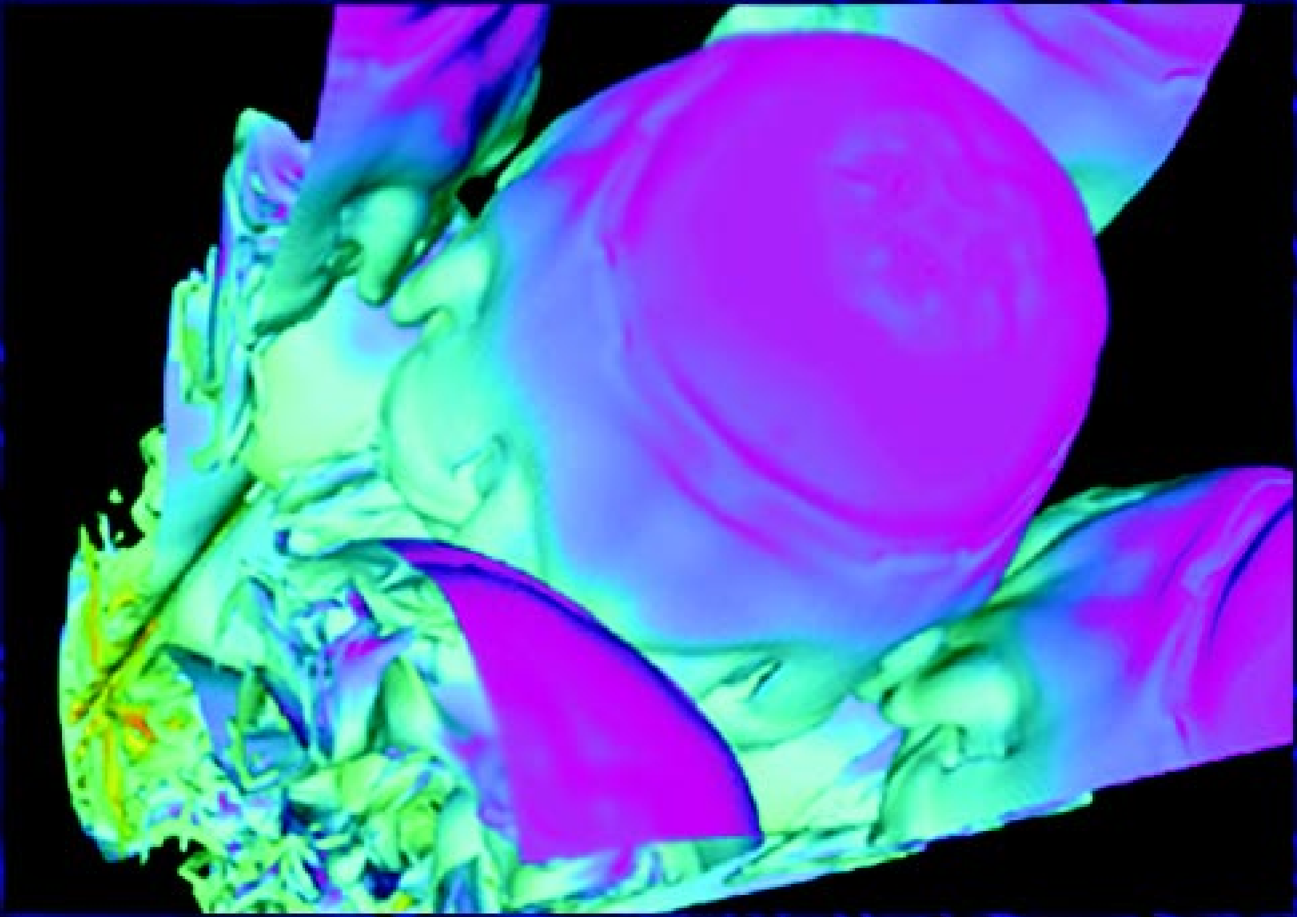
BITS

computing & communications news

Supercomputing allows for production 3-D parallel simulations

This figure shows a RAGE-code simulation of an initially perturbed interface between two materials after two shocks have crossed this surface. See inside Front Cover.



Los Alamos
NATIONAL LABORATORY

June/July 1999

ASCI code team members and visualization scientists analyze the animations created by the ASCI simulation data displayed on the Numerical Environment for Weapons Simulation collaboratory 15-ft-high by 10-ft-wide powerwall. For more information contact Robert Gurulé: phone 505-665-3538, or e-mail at rgurule@lanl.gov.

Front cover: **The result of a RAGE Richtmyer-Meshkov instability (Mach 1.2 shock) simulation of a single-mode, single-interface (SF6-air) setup. The original shock crosses the perturbed interface at time = 0, and the interface is then reshocked (time ~ 0.9 μsec) by the reflected shock. For more information, contact Robert Weaver: phone 505-667-4756, or e-mail at rpw@lanl.gov.**

Produced by the Computing, Information, and Communications (CIC) Division
Design: Donald Montoya and Wendy Burditt (CIC-1)
Managing Editor: Denise Sessions, denise@lanl.gov (CIC-1)
Illustration: Dave Delano
Photography: Gary Warren (CIC-9)
Printing: Imaging Services Group (CIC-9)

BITS Contributors' Board: Kimberlyn Mousseau, CIC-15; Gina Fisk, CIC-5; Geoffrey Brown, CIC-15; Kathleen Jackson, CIC-3; Kei Davis, CIC-19; Nikki Gaedecke, CIC-6; Octavio Ramos, CIC-1/NMT-DO; Noel Fletcher, CIC-1.

Table of Contents

New Ways for Managing Software Development Projects: A Walk on the Wild Side

by Don Willerton
Group Leader, CIC-6,
Customer Service

Software Project Management Series

This article is first in a three-part series in BITS exploring how you can keep a better handle on your software development project to ensure it doesn't spin out of control or face cancellation before the product's release. Because software development involves varying levels of uncertainty, that doubt precludes the use of traditional project management assumptions, processes, and tools. Of interest is the fact that recent developments are leading to different and better ways of managing software development projects.

Software Demands

Software projects are in a world of hurt.

Consider the following daunting statistics: (1)

- The majority of the 300,000 software development projects currently in progress in the U.S. will take longer, cost more than planned, and result in "out of spec" products that fail to meet user requirements. Many of them will be never be finished at all.
- More than half of the large projects (those involving more than a million lines of code) will be cancelled before a new type of software is ready for rollout.

So, what's the reason behind all the difficulties? As natural as it might seem to pin the blame on technology or the rapid change of technology, that's not what the studies say. Rather, a Defense Science Board report attributes project failures to "counterproductive management."(2) In contrast, another study comparing a successful project to an unsuccessful one pointed out that "internal management" was responsible for 60 percent of the cost improvement and 70 percent of the schedule improvement, as well as the overall success of the project. (3)

If both of those assessments are correct, internal management is the key to success or failure when it comes to developing software that survives beyond the idea and ultimately finds itself in the hands of users. Why, then, are software projects so hard to manage? Here is one of the main reasons:

The traditional project management processes, the tools supporting those processes and the ways of measuring and managing progress are almost useless for software development. Yet, the content of almost every "project management" course offering is dominated by traditional project management methods.

The bottom line is this: in learning how to manage projects, most software project managers are actually taught what will make them fail!

A Quick Look at Traditional PM Assumptions

A traditional project management class will make you familiar with these steps and outputs:

- Work breakdown structure,
- Work precedence network,
- PERT/CPM diagrams,
- GANTT charts,
- FTE loading profiles, and
- The overall role of the Project Plan.

Performing some of the typical classroom examples will underscore the objective of project management: to control the project to match the Project Plan. Defining the Project Plan, then, becomes the most important activity of the process, and it, by definition, is done completely before any technical work of the project is done. This reveals the assumption that almost everything about the project's activities is known and understood before the project is begun.

Consider some of the foundational assumptions that support the use of the Critical Path Method (CPM). (4) The CPM is used to quantify the key activities of any particular project whose durations directly affect the project finish date. The assumptions of CPM reflect the fundamental assumptions of the overall traditional project management philosophy.

With each assumption, imagine the reaction of the typical software project manager:

1. A project can be subdivided into a set of predictable, independent activities.

   Predictable? Independent? My requirements may not only significantly change during a project, but I have to actually do some of the project before I can know some of them! I have to discover or invent information in one activity that directly affects other activities in terms of content, duration, and cost!

2. The precedence relationships of project activities can be completely represented by a noncyclical network graph.

   Noncyclical? That means no DO LOOPS! But some activities are based on iteration, sometimes being repeated several times, before we can go on. I need a network graph that allows not only loops, but IF statements, to cover the things that I've learned don't have to be done.

3. Activity times may be estimated and are independent of each other.

   Estimated? I'm using a fluctuating workforce, a range of technical competence, new workstations, an object-oriented language that most of us are not familiar with, and I don't even have consistent documentation for our new supercomputer! How about if you just shoot me now?

4. The variance in the length of the project is assumed to be equal to the sum of the variances of activities on the critical path.

   An overall delay is the sum of the critical path delays? I have many critical issues and players that influence my project on a daily basis. Any statistical feeling of whether I'm ahead or behind of schedule doesn't guide my decisions.

5. The duration of an activity is linearly (and inversely) related to the cost of resources applied to the activity.

   In theory, this means that if I have double the number of people on an activity, the activity will be done in half the time. I can remember Fred Brooks explaining that adding more people to a software project will eventually have a negative effect, so maybe this works for widget-making, but not for me.(5)

Why Is Developing Software So Hard?

Software development is a process filled with uncertainties. The progress of a software development project may rely on information being discovered or created in the process of performing the project. With projects with high uncertainties, the ideas of estimating, planning, sequencing, and progress seem out of sync with reality.

In *The Project Survival Guide*, Steve McConnell put it this way(1):

"If you want to understand what software development is all about, you need to understand that the project team has to think through and make all the decisions in one stage before it can know enough about the next stage even to estimate the work involved in it."

One can guess at a few of the things that will happen if you use traditional management methods for projects with high uncertainty:

- Customers and sponsors may have false expectations in start dates, end dates, activity and project durations. These expectations can carry over into cost and specifications. It's the usual "being held to a schedule that was created before the project was defined!"
- Traditional project management will not account for "rework"—the work that will have to be done over because of bugs, changes in requirements, or design changes. McConnell says that rework may account for *80 percent of all of the work* of the project!
- The project team will spend useless effort on traditional overhead activities, like reporting, documentation, audits, evaluations, and administrative reviews, with no value added. Upper management and stakeholders have little confidence in project status reports that have phrases like "well, we think that we'll try this for a while ..." and, yet, *it may be exactly the right thing to do*!
- Software project managers will face big dilemmas with customers and sponsors who don't recognize or aren't familiar with the difficulties of projects that involve high levels of uncertainty, exploration, invention, or discovery. It may be especially difficult when the only thing they do understand is a GANTT chart.

The Wild Side: Nontraditional Project Management

Because none of these difficulties are surprising to those in the software business, some software thinkers, researchers and practitioners have developed different, nontraditional approaches to project management. Many of these approaches specifically address the differing levels of uncertainty in projects, as well as prescribing good practices that raise the determinacy of project performance, better requirements matching, lower project cost, faster development times, higher quality, and other factors.

Coming in the next issue of BITS, we'll look specifically at some of the new approaches in "Adjusting for Reality: Mitigating Uncertainty in Projects."

References

(1) S. McConnell, *Software Project Survival Guide* (Microsoft Press, Redmond, WA, 1998).

(2) N. Brown, "Industrial-Strength Management Strategies," *IEEE Software* **13**, (no. 4) 94-103 (1996). Also, at http://www.spmn.com/whatsnew.html#evm

(3) K. Cooper, *The Four Failures: Systemic Sources of Project Problems* (Paper for Pugh-Roberts, Boston, MA, 1998).

(4) J. D. Wiest and F. K. Levy, *A Management Guide To PERT/CPM* (American Management Association, New York, 1977).

(5) F. P. Brooks, Jr., *The Mythical Man-Month* (Addision–Wesley, Reading, MA, 1975).

Scientific Software Process Engineering: Finding the Poetry in Process

by Maysa-Maria Peterson-Schnell, Scientific Software Process Engineering Project Leader CIC-12, Software Design and Development Group

The Information Technology industry has long viewed software development as a temporary crisis working in fire-fighting mode to deliver product to market as quickly as possible, but this is changing. This crisis often meant that software was being developed and delivered with numerous known defects and/or limited functionality. This is the realm of "good enough" software not unlike the parable of the two men in the woods who meet a bear and take off running. One man says to the other, "Why are we running? We can't outrun a bear." His friend turns to him and replies, "I don't have to outrun a bear. I only have to outrun you."

Risk-taking in a Competitive Environment

As awareness of the business importance of this "temporary crisis" and the significance of software and its impact increases, more and more corporations are looking toward identifying what core competencies they require to meet their business objectives. They are discovering that one of their biggest risks lies in their software and their software development practices. Out of this realization grew the software engineering subspecialty area called Software Process Engineering (SPE).

SPE requires knowing what competencies your organization needs and building a framework for strategic improvement. Software processes can be organized in a number of ways and there are various models from which you can build your framework for process improvement. There's plenty of room for improvement when it comes to software development for the majority of corporations that invest significant dollars to produce software products for internal or external users. The Software Engineering Institute's figures indicate that at least 67 percent of all software development companies are operating at Level 1 of their Capability Maturity Model (CMM). Level 1, or the initial level on the CMM scale, is described as ad hoc and occasionally chaotic. Few processes and standard practices are defined or understood and success is often dependent upon the heroic efforts of a few people. At the initial level, the organization typically lacks sound management practices and thus does not provide an adequately stable environment for developing and maintaining software. When organizations lack this level of management, fire-fighting is the mode of operation during a crisis, and the development focus is diverted to coding and testing, undermining any efforts to sustain an engineering practice. These software development companies work in many domains from financial database applications to scientific R&D but much can be learned from sharing lessons learned across domains, organizations, programs, and projects.

Creativity Enhanced by Process

I am a poet as well as a software engineer – a right-brain creative activity. Processes and plans are left-brain activities but this alone shouldn't limit me in attempting to improve my personal software development practices. The point is that the process and its definition must be flexible, adaptable, portable, modular. Sounds like a definition of truly "good," enduring software product. But being a poet, I want to find the poetry in the process. In other words, process shouldn't limit creativity but rather enhance it. I rarely sit down and say, "I'm going to write a sonnet, or a renga." I usually sit down to write and eventually the form will appear. Granted, sometimes I have to write something in a particular style but that's not how the poem is formed – with the style in mind. A bright idea becomes transformed into an object. Software development is the same.

If I was to apply process improvement to my poetry writing what would I do? First I would take a look at my current practices. Assess them. I might want to look at when I write and what comes out of those sessions. When am I most productive? What helps me be more productive? Wat is the quality of the poems written? Which get published and which don't? I might define my process, read other books on writing practices, refine how I do things, perhaps try new methods to get the words flowing, and then see how I feel about the new work. Do these new poems get published, awards, or payment?

Assessing and Improving Current Practices at LANL

There are a couple of pilot projects underway at LANL to assess and improve current software development projects in various key process areas. The intent of the Scientific Software Process Engineering (SSPE) project is to assist projects in meeting their software quality and development process improvement goals. The project was created primarily to support X-Division Accelerated Strategic Computing Initiative (ASCI) projects and secondarily to support lab-wide projects with the implementation of their software process engineering and quality assurance programs. It also is involved in tri-lab activities for ASCI. The project provides training, educational opportunities, consulting services, software process assessments, software improvement planning, process improvement metrics, and is establishing a database of local best practices. It is a clearinghouse of information and resources for software process improvement. The SSPE project's main goals are to:

- work with organizations and projects to help assess and define their current software development practices,
- refine and enhance their current practices,
- help implement new best practices that have a proven track record and return on investment, and

- monitor process improvement to guarantee that corporate strategic goals are being met, creativity is facilitated, and projects deliver high-confidence, enduring software products.

Software Verification

In Patrick Roache's latest book on verification and validation, he shares a personal anecdote illustrating the importance of software verification – a software development process. He discusses a particular project that he was involved in where he "ostensibly benchmarked the code by comparison" to verify the code's correctness. He released it to colleagues for preliminary use, and upon execution they discovered that it "did not converge to the correct answer." He decided to take a look at the code and discovered a defect. His conclusion to this story was that an inadequate amount of testing was performed such that this defect went undetected.

It was fortunate that Mr. Roache was able to find this defect. It is unfortunate that he attributes its discovery, or previous lack of discovery, to the wrong process. My point is that, because the code was "ostensibly benchmarked" and the defect went undetected, it took another software process–code inspection, a type of peer review process–to identify the defect. He is correct that ostensible testing did not detect it but perhaps had he inspected the code previously in a peer review process, he may have saved himself a lot of time and energy.

Rework: Killer of Productivity

Roache's story is another example of a useful software process that has been identified to give code teams the greatest return on investment while greatly reducing rework, which is the killer of productivity in the software industry. Sommerville states that of defect types found they are typically these types listed by percent of occurance:

- Incorrect facts, 49%
- Omissions, 31%
- Inconsistency, 13%
- Ambiguity, 5%
- Misplaced requirement, 2%

A comparison done by Pressman relates defects to cost in hypothetical projects that match similar criteria. (Data collected during real projects shows similar trends.) Using what Pressman calls the defect amplification model, the earlier in the software development process that reviews are begun, the greater the productivity gains are achieved. In the model, each step is assumed to uncover and correct a percentage of all incoming defects without introducing any new defects (a simplistic view). Figure 1 below shows that the total cost for development and maintenance is three times greater when no reviews are performed for the same hypothetical project. The increase in cost is directly a result of the amount of rework required to fix mistakes that could have been caught much earlier.

No Reviews Conducted

Phase Error Found	Number of Errors	Unit Cost per Error	Total Cost
Code	22	6.5	143
Testing	82	15	1230
After release	12	67	804
			Total 2177

Reviews Conducted

Phase Error Found	Number of Errors	Unit Cost per Error	Total Cost
Design Review	22	1.5	33
Code	36	6.5	234
Testing	16	15	315
After release	3	67	201
			Total 783

Figure 1. These tables illustrate that the total cost for development and maintenance is three times greater when no reviews are performed for the same hypothetical project.

Being that the cost of defects found later in the software development process increases exponentially, it would be beneficial to develop processes and practices to identify and eliminate defects early in development. Sommerville also compares the relative cost of defects found within the following life-cycle phases listed by number of defects:

- Requirements, 0.1 – 0.2
- Design, 0.5
- Implementation, 1.0
- Unit Testing, 2.0
- Acceptance Testing, 5.0
- Maintenance, 20.0

SSPE Pilot Process Improvement Activities

Code inspection is one example of a peer/personal review process. Peer reviews should also be applied in the earlier phases of software development and to all software work products such as Software Requirements Specifications (SRS) and Software Design Descriptions (SDD). These are some of the many processes that the SSPE project is evaluating and piloting for the scientific software development community. SSPE is also working on project management and tracking, requirements management and tracking, software verification and validation, software testing techniques, software configuration management, and software quality assurance.

In addition to software process improvement, SSPE also sponsors the Software Engineering (SE) Lecture Series and the SE Courses Series. Some of the future speakers planned for the SE Lecture Series include Tom DeMarco (risk), James Bach (testing), Greg Wilson (project-long strategies), and John Beidler (reuse). The SE Course Series includes Software Risk Management, Python OO Program Development, and Software Testing Techniques. SSPE is sponsoring *The Software Quality Forum* to be held in Santa Fe in May 2000. This large conference will focus on software quality issues as it affects development, verification and validation, testing, CASE Tools, project management, and requirements traceability for scientific software projects.

Applying Process Improvement

If I look again at writing poetry and in applying process improvement, I realize that I spend a lot of time fixing my typewriter because I notice defects. Every time I type a “B” it looks like a “P”. I conclude that I need a new typewriter. Perhaps I will modernize my process and purchase a word processor. But without looking at how and where I am spending my time it is easy to waste time fixing the typewriter and hard to recognize what improvements I can make.

Because we want to be productive and deliver products on time, it is hard to keep as flexible, adaptable, portable and modular as we need to be to improve the process of software development. But if we save time and money by reducing rework, we’ll have more time for creative pursuits, like writing poetry. For process engineering, it’s all a matter of finding out what processes work best in what environments in order for people to meet their strategic and quality goals. For me, it’s about finding the poetry in the process.

References

Paulk, Mark, et al., “Capability Maturity Model for Software Version 1.1,” (Technical report, SEI–93-TR024, Software Engineering Institute).

Roache, Patrick J.*, Verification and Validation in Computational Science and Engineering* (Hermosa Publishers, Albuquerque, 1998).

Pressman, Roger S., *Software Engineering*, Fourth Ed. (McGraw-Hill, 1997).

Sommerville, Ian; *Software Engineering* (Addison-Wesley, Reading, MA, 1996).

Lecture Review ACL Seminar Series: Performance and Scalability Analysis of Applications on Teraflop-scale Architectures

by Michelle Pal,
Graduate Research Assistant
CIC-19, Scientific Computing Group

On March 15, 1999 Adolfy Hoisie gave the talk entitled "Performance and Scalability Analysis of Applications on Teraflop-Scale Parallel Architectures," one of the ongoing ACL Seminar Series. This talk reported work performed by Adolfy Hoisie, Olaf Lubeck and Harvey Wasserman of the Parallel Architecture team, Scientific Computing Group CIC-19, Los Alamos National Laboratory.

The outline of the talk was:

- Parallel Wavefront Algorithms;
- Deterministic Particle Transport: Resource requirements;
- Performance and Scalability Modeling: MPPs and clusters of SMPs;
- Performance Predictions for Teraflop-scale Architectures;
- Conclusions.

Parallel Wave Algorithms

Adolfy introduced the parallel wave algorithms with visualization of the wavefronts in 1-D, 2-D, and 3-D. Wavefront techniques enable parallelism by splitting the computation into segments and pipelining the segment through multiple processors. A good balance between computation and communication cost makes the wavefront computations a good choice for SWEEP3D, a deterministic particle transport code taken from the DOE ASCI benchmarking suite. SWEEP3D is used for a neutron transport simulation of a time-independent, multi-group, inhomogeneous Boltzmann transport equation. Approximately 50 to 80 percent of the execution time of simulations on the DOE systems are dedicated to deterministic particle transport, and this high percentage may expand on future generation systems. The estimated memory requirement for 10^9 spatial computational cells on a hypothetical 100-petaflop machine is 40 terabytes. Estimated execution time for a 0.1 msec grind time per phase-space cell is 480 hr per time step, whereas for a 30-hr runtime goal the minimum execution speed is 160 teraflops, assuming perfect scaling.

A basic pipeline model was introduced for a 2-D processor domain of P_x x P_y size. The pipeline model comprises both computational and communication stages and a formula for the number of steps required. The computation cost at each step and the number of communication stages is given in closed form.

A pseudo-code for the pipelined wavefront abstraction was given; this is the essence of the approach and easy to understand.

```
for each octant
  for each angle-block
    for each z-block
        receive east
        receive north
        compute sub-grid
        send west
        send south
    end for
  end for
end for
```

Analysis of the contributions of computation and communication validated the model with the performance data from SWEEP3D on three architecturally disparate and widely used machines: the IBM SP2, CRAY T3E, and SGI Origin 2000, each with up to 500 processors. Three validation regimes were considered: number of sweeps equal to 1, contributions of P_x+P_y when the number of sweeps is comparable, and number of sweeps much greater than P_x+P_y. For each of these cases the problem size was such that the computational cost is zero, comparable with, or greater than the communication cost, respectively. The agreement of the model with the measured data was excellent for all three validation regimes.

Pipeline with Bottlenecks

For clusters of SMPs, a "Pipeline with Bottlenecks" model was presented. This model shows that for a 1 billion-cell problem, in marked contrast to most large simulation problems, the bottleneck is the single-node computation speed, not the communication cost.

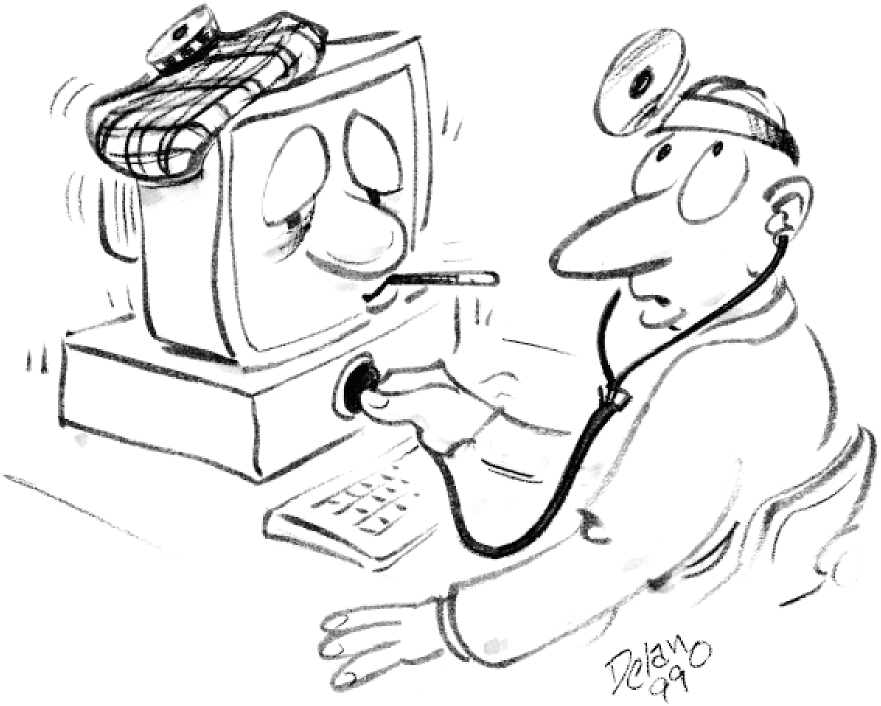
Generally speaking, for good performance analysis the following factors should be taken in account: problem size, topology and size of the processor cluster, communication and computation parameters, and problem-blocking size. The model introduced is parametric in all of these parameters.

Several plots of time versus number of processors were presented, followed by an estimate of SWEEP3D performance on a hypothetical future-generation 100-teraflop system, as a function of Message-Passing Interface (MPI) latency and sustained per-processor computing rate.

Making such a prediction is problematic because of the difficulty of making accurate estimates of the performance of future systems, managing the parameter space of SWEEP3D, and prediction of the size of the problem of interest in the future.

Concluding Remarks

A few concluding remarks ended the talk. Adolfy stressed that the key to a meaningful point design study is the application-architecture mapping and that such a study should address a specific workload. Back-of-the-envelope performance prediction is difficult because of the complexity analysis in a multidimensional performance space. As an example, for the 1-billion-cell problem, a 30-hr goal was set but the most optimistic technological estimates fall short by a factor of two. At such large scales the engineering of the systems and the applications need to be performance based. The model introduced is scalable for parallel multidimensional wavefront calculations on LogGp machines. This model enabled meaningful performance analysis of a deterministic transport code on a hypothetical parallel system for solving ASCI problems, in particular SWEEP3D applications.



Reference

1. A. Hoisie, O. Lubeck, H. Wasserman, “Scalability Analysis of Multidimensional Wavefront Algorithms on Large-Scale SMP Clusters,” In *Proceedings of IEEE Frontiers ‘99*, (IEEE, Annapolis, MD, February 1999).

Major 3-D Parallel Simulations

by Robert Weaver
Technical Staff Member,
X-TA, Thermonuclear Applications Group

One of the outstanding challenges in hydrodynamics simulations is the accurate modeling of compressible material mixing. The relevant length scales necessary to properly model this physical process of material mixing vary across many orders of magnitude. To simulate these types of hydrodynamic phenomena the Los Alamos Crestone project team is using the Eulerian Continuous Adaptive Mesh Refinement (CAMR) code RAGE. The Crestone project is lead by co-team leaders Michael Clover (X-CI) and Bob Weaver (X-TA) and senior code architect Mike Gittings (X-CI and Science Applications International Corporation). The project is developing both unclassified and classified parallel codes for the DOE Stockpile Stewardship Program. The unclassified RAGE code is a one-dimension (1-D), two-dimension (2-D) or three-dimension (3-D) parallel, Eulerian, radiation hydrodynamics code based on a high-order Godunov method that features the CAMR technique.

Parallel Computing: Transforming our Outlook on Computing

Whereas there have been numerous simulations of hydrodynamic instabilities in 2-D, for both simple and complex perturbations, there have been relatively few 3-D simulations of instability growth. The computing power at Los Alamos increased by more than two orders of magnitude with the delivery of the Accelerated Strategic Computing Initiative (ASCI) Blue Mountain supercomputer in November 1998. This tremendous increase in capability has transformed our outlook on computing. Simulations that would have taken a lifetime to complete are now finished routinely in a fraction of a year.

Parallel computing has allowed Los Alamos scientists to accomplish more computing in the last year than was physically possible throughout the first 50 years of computing at Los Alamos!

Specifically, the number and complexity of completed 3-D hydrodynamic instability simulations has skyrocketed. The primary purpose of these calculations is to help validate the RAGE code 3-D hydrodynamics. Figure 1 shows a Rayleigh-Taylor mixing of two fluids, which is an example of one such RAGE simulation. By combining all of the parallel 3-D production runs by members of the Crestone team over the course of the last year, we have run the equivalent of several centuries of continuous, single-processor Blue-Mountain computer time!

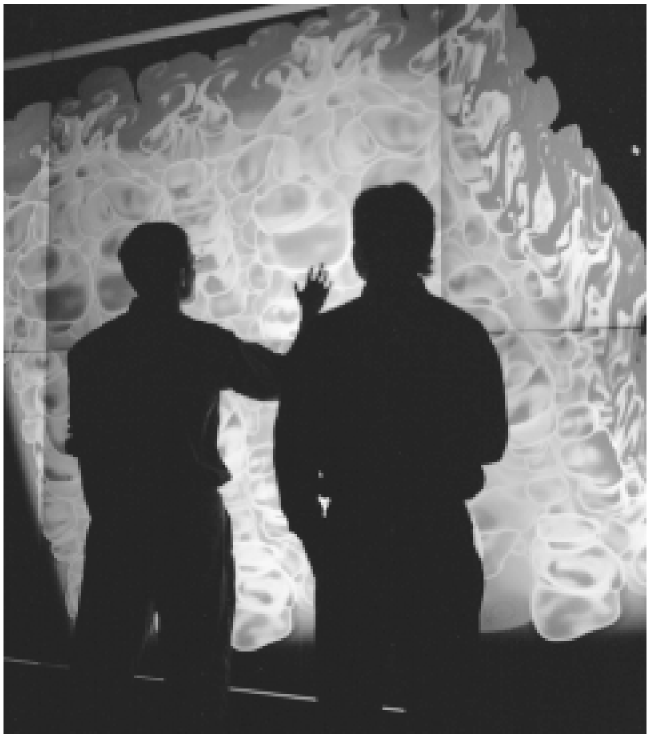


Figure 1. An example of a Rayleigh-Taylor mixing simulation as calculated by the RAGE code. This image is being viewed on the Los Alamos Powerwall in the visualization theater, or "collaboratory."

The initial goal of the Crestone project has been met by producing a 3-D parallel version of the CAMR RAGE code. The primary emphasis in this development has been scalability, usability, portability and reliability, and less effort has been spent on optimization. As a result, we are now able to run real problems on a variety of platforms, including the SGI/CRAY Origin 2000 Blue Mountain supercomputer at Los Alamos, the Intel TFLOPS ASCI red supercomputer at Sandia, and the IBM SP2 supercomputer at Livermore. Most of our experience has been on our local Blue Mountain machine. Scalability testing has been an ongoing part of our development of this parallel version of the RAGE code. This scalability test involves running a series of simple, 3-D hydrodynamic simulations to assess the speedup of the code as more processors are used. This scaling study uses a constant workload on each processor (constant number of zones per processor; 13,500 cells per processor is a typical number.

As the number of processors increases we expect that the speed of each run should increase linearly with the number of processors. The object of this exercise is to run very large jobs (thousands of processors and terabytes of memory) as fast as we can run smaller jobs (tens of processors and megabytes of memory). Our target is scalable performance to greater than 6000 processors on a 1-billion-cell CAMR computational cell problem that requires hundreds of variables per cell, multiple physics packages (e.g., radiation and hydrodynamics), and implicit matrix solves for each cycle.

Figure 2 shows the current results of this scaling study on a variety of supercomputers. Linear scaling is shown as the light blue and red dotted lines. The fundamental parallel algorithms of RAGE [using message-passing interface (MPI)] are proven to scale on both the ASCI Red machine and the SGI T3E 1200. These same algorithms, which were run on the Blue Mountain architecture, serve to highlight avenues for improvements to the hardware (e.g., interconnects) and software (e.g., MPI library) of this class of parallel supercomputer. Although we can run on the Blue Mountain supercomputer using all the 48 symmetric multiple processor boxes and over 6000 processors available, these results show that there is an order of magnitude decrease in scalability at 4000 processors, and a catastrophic failure over 4000 processors. One of our top priorities for current work is to collaborate with SGI onsite analyst Howard Pritchard to understand and improve this scaling for the Blue Mountain computer.

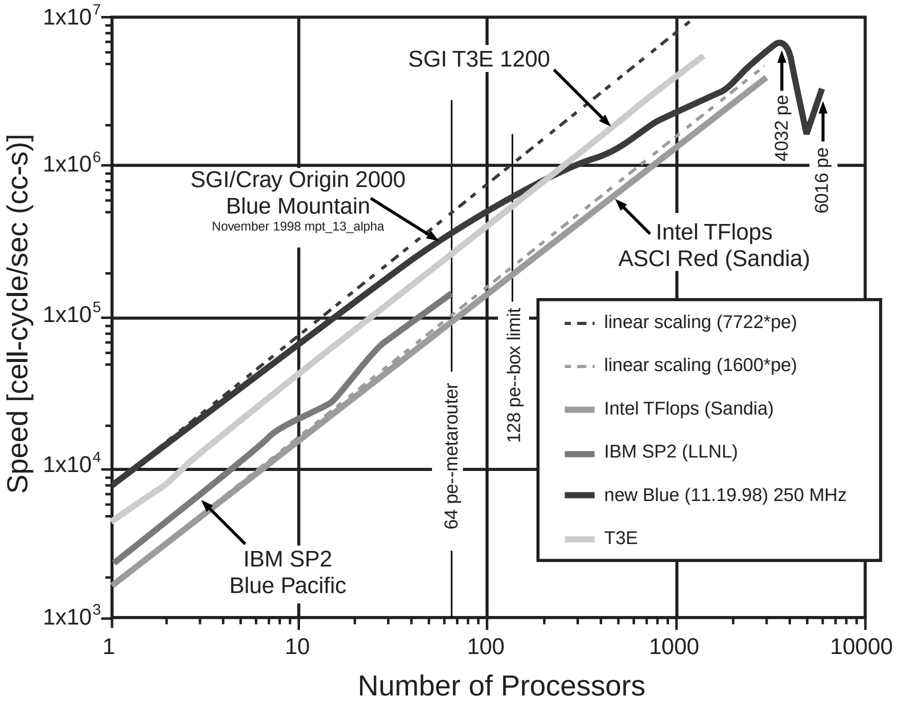


Figure 2. A demonstration of the scalable nature of the parallel algorithms developed by the RAGE code. The figure shows the speed of the calculation (cell-cycles per second) as a function of the number of processors. The blue curves (dark blue = Blue Mountain, light blue = Blue Pacific) are results from SMP machines. The other lines on this figure are results from MPP machines. The red curve is from the Sandia ASCI Red machine, while the magenta curve is from a SGI T3E 1200 machine that we had access to during December 1998. (ED: If not viewing in color, please note arrows and labels that identify which supercomputer ran code.)

Validating RAGE Code: Simulation of Hydrodynamic Instability

Simulating hydrodynamic instability requires an initial pertubation of the surface between two adjacent fluids. This pertubation may be simple, such as a single mode or wavelength, or may be complex, such as would result from machining, or tooling processes, i.e., multi-mode. The Crestone team had devoted a major effort to simulating the growth of hydrodynamic instabilities, particularly 3-D simulations. All three major types of hydrodynamic instabilities have now been calculated with the RAGE code on the Blue Mountain machine:

Rayleigh-Taylor (RT), Richtmyer-Meshkov (RM) or shock-generated instability growth, and Kelvin-Helmholtz (KH) or shear-flow instability. Many types of simulations have been performed with RAGE: from 1-D X-Division analytic test problems, to 2-D simulations of laboratory experiments, to 3-D single-interface, single-mode and multiple-mode instability simulations. The RAGE code is also extensively used in the design of Above Ground Experiments (AGEX) to obtain new experimental data to contribute to the validation of the code. When compared to the relevant experimental data or theory, one of the purposes of these instability simulations is to contribute to the validation of the code.

__Figure 3. The author in front of an image from a RAGE simulation of single-mode single interface RMI on the Powerwall.__

Results Simulating 3-D Richtmyer-Meshkov Instability Growth

The RAGE code was originally designed as a shock code for solving problems such as those encountered in the Defense Nuclear Agency (DNA) water shock program. Many 2-D Richtmyer-Meshkov Instability (RMI) simulations have been performed over the years, and the results compared to both other code results and experimental data. Many of these RAGE simulations have been published in various journals and conference proceedings. (For more information, contact the author at 505-667-4756 or e-mail rpw@lanl.gov.) The delivery of the Blue Mountain supercomputer has allowed Los Alamos scientists to routinely perform 3-D RMI simulations that allow us to develop better theories for the true rate of this instability growth.

Figure 3 shows the result of a RAGE simulation of a single-mode single-interface (SF6-air) RMI (Mach 1.2 shock) that has been shocked at time = 0 and then later re-shocked (time ~ 0.9 msec) by the reflected shock. This figure shows the isodensity surface (0.004 gm/cc), colored by speed of the material at a time of 1.3 msec. The reflected shock traverses the interface from the direction opposite that of the original shock and results in an inversion of the original bubbles and spikes. Figure 4 shows the same simulation from the opposite perspective, viewing the inverted "bubble side" of the isodensity surface. This figure depicts the surface 2 msec after the simulation. Quick-time movies of these simulations are available for viewing in the author's office.

We are pleased with the progress on validating the RAGE code for 2-D and 3-D RMI and believe that the macroscopic features of these flows, e.g., amplitude and growth rate, are accurately simulated by RAGE. However, notice that there is a great increase in the small-scale features after reshock. We are working with J. Kamm and W. Rider of X-HM to assess how much of these small-scale features are reasonably simulated by RAGE.

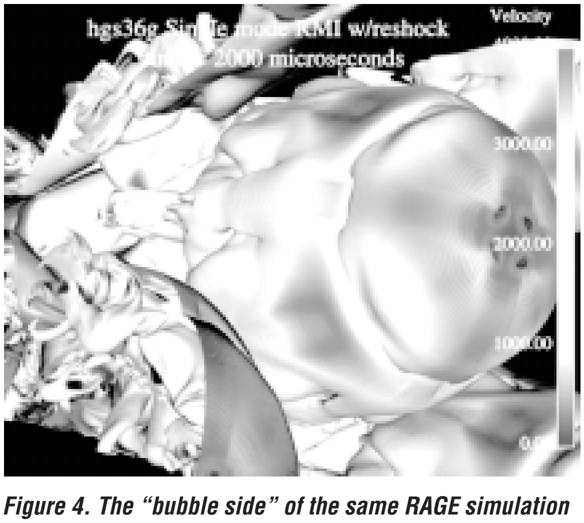


__Figure 4. The "bubble side" of the same RAGE simulation shown in Fig. 3 at a time of 2 msec. Note the amount of small-scale structure in this simulation. (Insert: hgs 36 g single mode RMI w/reshock time = 2000 microseconds)__

Extensible HTML: Preparing for the Next Phase of Web Markup

by Tad Lane, Information Architecture Standards Editor, CIC-1 Communication Arts and Services

A little over a year ago, I wrote a BITS article that included five tips on how to write HTML 4.0 so that it would be as close to compliant with XML as possible. In March 1999, the World Wide Web Consortium (W3C) moved the Working Draft of its XML-compliant version of HTML 4.0 to "final call" status, which presumably means that it is relatively stable and we should very soon see it emerge as a Proposed Recommendation. The specification for "Extensible HTML 1.0" (XHTML) goes well beyond my five tips from last year and includes not only the changes to HTML 4.0 that are needed to make it fully XML compliant, but also tips on how to write XHTML so that older browsers can render it correctly. In light of the new specification, now seems like a good time to take a look at the shape of things to come.

The biggest difference between last year and this year is that XHTML 1.0 has in fact appeared and can in fact be used to write documents that current browsers can understand. To prepare for this article, for example, I wrote several XHTML documents and tested them out under various browsers. The only problems I've encountered so far have been minor differences in rendering, with nuisances like certain instructions being displayed on screen when they should be hidden. Small problems, all in all, and none appearing in fourth- or fifth-generation browsers.

In short, although XHTML is not yet ready to become our new Laboratory standard, it is plenty mature enough for us to start experimenting with it and learning how to use it within the constraints of our current environment. It is still too early to take full advantage of XHTML, but it is not too early to familiarize ourselves.

Differences between XHTML and HTML

Although most of HTML 4.0 translates directly into XHTML, there are a few things that need to be done differently.

1. Begin the XHTML document with the following line:

   <?xml version="1.0"?>

This states that the document is an XML document and can contain other attributes to specify, for example, which language it is written in.

2. Always include a DOCTYPE declaration (which was optional in HTML 4.0).

   This must be the first line after the "<?xml?>" line, before the opening <html> tag, and must be one of the following:

   <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "xhtml1-strict.dtd"

   <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "xhtml1-transitional.dtd">

   <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "xhtml1-frameset.dtd">

As with HTML 4.0, the "Strict" DTD (Document Type Definition) excludes all the deprecated markup that W3C expects to phase out (such as the <font> tag being phased out in favor of style sheets); the "Transitional" DTD is somewhat looser, allowing the deprecated markup; and the "Frameset" DTD is used for pages with frames.

Note: To get the DOCTYPE declaration to work with certain XML checkers, I've found that I needed to copy the ".dtd" and ".ent" (entity set) files from the W3C XHTML 1.0 specification over to local files. This appears to be because the checkers are not correctly interpreting "-//W3C//DTD", so I hope it is not a permanent problem. For now, though, local copies seem to work as a workaround.

3. The XHTML document must specify an XML namespace (xmlns) with one of the following in the opening <html> tag (depending on which DOCTYPE is selected):

   <html xmlns="http://www.w3.org/Profiles/xhtml1-strict">

   <html xmlns="http://www.w3.org/Profiles/xhtml1-transitional">

   <html xmlns="http://www.w3.org/Profiles/xhtml1-frameset">

4. Tags must be properly nested. The following is an example of incorrect nesting: "<b><i>Bold italic</b></i>". A correct example is "<b><i>Bold italic</i></b>". This is also required in the HTML specification, but current browsers frequently overlook errors.

5. All tags and elements must be in lowercase. Although "<P>" and "<p>" are identical in HTML 4.0, only "<p>" is acceptable in XHTML.

6. End tags are always required for non-empty elements (i.e., any tag that contains content outside the tag itself). For example, although this was optional in HTML 4.0, list items must have closing tags in XHTML:

   ```
   <li>list item</li>
   ```

7. All attribute values must be quoted in XHTML, even those for which quoting was optional in HTML 4.0:

   ```
   <td align=”center” valign=”top”
   width=”150">table cell</td>
   ```

8. All attributes require values in XHTML. For HTML 4.0 elements that did not have a value, use the attribute name itself as the value. For example, “<td nowrap>” in HTML 4.0 becomes the following in XHTML:

   ```
   <td nowrap=”nowrap”>
   ```

9. Use “ />”, including the space, to close empty elements (i.e., elements such as line breaks or images where all content is included within the tag):

   ```
   <img src=”source.png” />
   ```

Note that the alternative approach of using a separate closing tag (“<br></br>”) is discouraged because of unpredictable browser behavior. Also note that this closing is required for empty tags within the <head> section, such as “<meta />”, but not for the “<?xml?>” or “<!DOCTYPE>” lines.

10. Wrap scripts or style element within “CDATA” to avoid having those characters expanded by the XML processor. This is done in XHTML as follows:

    ```
    <script>

    <![CDATA[

    script content

    ]]>

    </script>
    ```

For HTML browsers, however, the CDATA opening and closing needs to be further commented out in order for the script to operate:

```
<script>

<!-- <![CDATA[

script content

// ]]> -->

</script>
```

Note that I’m not absolutely positive that the above construction is really “good” XML, but that it makes things work for the browsers and is correctly interpreted by XML checkers. Because of some of these types of potential problems, a preferred approach is to use external scripts and style sheets, especially if they include “<“, “&”, or “]]>”.

File Types and Filename Extensions

The document itself can be named with one of three filename extensions: “.xml”, “.html”, or “.xhtml” (provided the server is configured to serve the extension, as in “text/html”): The “.xml” extension specifies that the document is an XML document, but it loses all XHTML-specific rendering (e.g., a <h1> header is not rendered in a larger font unless a style sheet has been used to specify it).

The “.html” extension makes the browser think that the document is a regular HTML document, which leads to the document being rendered as if it were HTML. This loses some of the XHTML-specific rules (such as requiring “ />” to close empty tags), but it allows current browsers to effectively read the document (provided that the tips below are followed).

The “.xhtml” extension specifies that the document is an XHTML document, which may eventually lead to the richest use of the XHTML capabilities. Current browsers do not yet support the XHTML file type, however, and it is not yet certain that the “text/xhtml” media type will in fact be registered.

In today’s environment, the “.html” extension seems the most useful compromise.

Tips for Writing XHTML for Current Browsers

The XHTML specification also offers a series of guidelines for writing XHTML in a way that will be correctly interpreted by current browsers and other user agents (such as Braille pads or voice synthesizers). For illustrative purposes, I’ll present some of these tips here (those that strike me as most useful or which I haven’t already included above). For readers who are serious about testing out XHTML today, though, I strongly recommend referring to Appendix C of the XHTML specification itself.

1. Pay particular attention to points 9 and 10 above, involving the space in “ />” to close empty elements and the use of external scripts and style sheets whenever they contain “<“, “&”, or “]]>”.
2. Use “ />” only to close elements that are defined to be empty, not those that just happen to be empty. For example, “<hr />” is correct, but “<td />” for a table cell that happens to have to content can cause problems. Instead, use “<td></td>”.

3. Line breaks and multiple white spaces in attribute values can cause problems and should be avoided. For example,

   <meta name="description" content="brief description

   of document" />

   would be more consistently interpreted if the line break occurred before the start of attribute:

   <meta name="description"

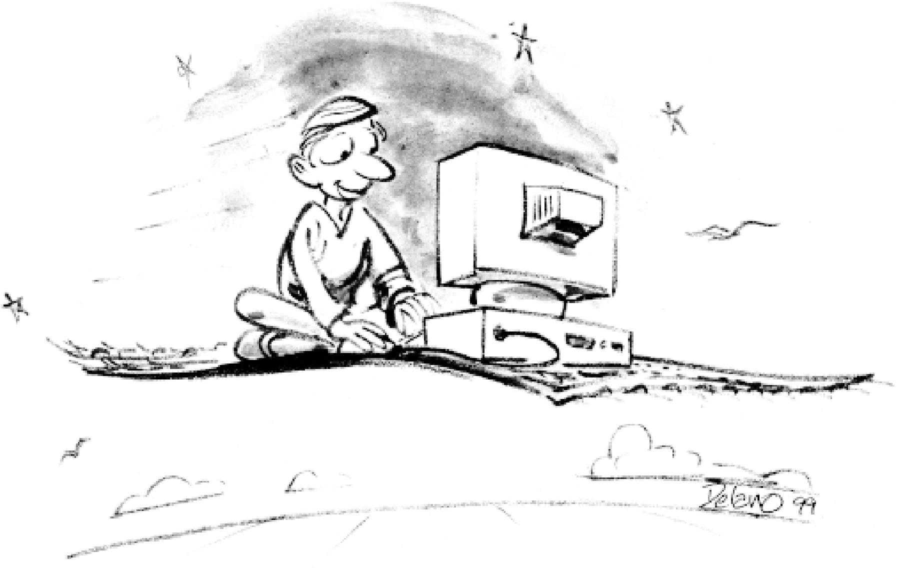
   content="brief description of document" />

4. As always, test on multiple browsers on multiple platforms to make sure the document is acceptably rendered. For example, I've found that some older browsers display the opening "<?xml?>" line, which qualifies as a nuisance but not necessarily unacceptable.

For Further Information

For the most current version of the XHTML 1.0 specification, see the W3C Technical Reports and Publications page at http://www.w3.org/TR/. For more background on XML, see the March 1998 BITS article "Extending Web Documents: Getting Ready for XML."

For other links related to XHTML and some sample XHTML pages, see the Information Architecture Web Team page at http://www.lanl.gov/projects/ia-lanl/area/web/. For more information about the IA Project in general, see our project page at http://www.lanl.gov/projects/ia/.



Virtual Private Network

by Tad Lane, Information Architecture Standards Editor, CIC-1 Communication Arts and Services

You are in a crowded noisy Cairo street market. Vendors hawk their wares, haggle with customers. Customers shout among themselves, jostle for position, bump into you, "invade your space" (in California lingo). Some hundreds of yards away, down an equally crowded side street, I am there, too. We can't see each other.

Then you speak in a normal tone of voice, and I hear you, and nobody else can. And I speak, and you hear me, and nobody else can.

The crowd is the Internet. We talk to each other through a "tunnel." And there can be others besides you and me who join our talk, with all of our communications among ourselves equally protected from eavesdropping by others.

And that's a Virtual Private Network (VPN)

Here at the Laboratory, in tandem with the unclassified network security firewall we recently implemented, we have also established VPN capabilities that enable remote users to become, in effect, part of the Unclassified Protected (blue) Network. If you're out visiting Berkeley, for example, the VPN enables you to connect to the university's network, tunnel into Los Alamos through the Internet, and securely perform unclassified computing just as if you were sitting at your desk in the Admin. building. This article provides an overview of VPN basics, along with discussion of the encryption we use and VPN limitations and applications. At the end of the article is information about how to obtain preconfigured VPN client software, along with links to additional information. Because our VPN capabilities are relatively new, there are still some limitations to it. As of this writing, for example, VPN client software is only available for Windows 95/NT, though the Macintosh client is expected this summer. It looks like we'll need to wait for the technology to mature further before UNIX clients become available. Because of those limitations, other services are provided for external users without VPN software to access the Laboratory networks. (The most notable of these are SSH (secure shell) and Kerberos, which are more fully described in the Information Architecture's "SSH/Kerberos FAQ" at http://www.lanl.gov/projects/ia/stds/lanl/ssh-faq.html.)

VPN Basics

As illustrated in Figure 1, the VPN works by establishing a secure tunnel between a remote client machine and the Laboratory network.

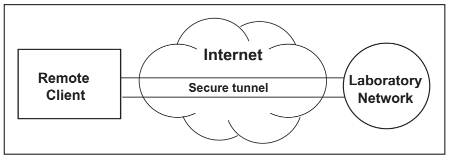


Figure 1. Basic VPN Connection

This is similar to a dial-up modem connection in that it acts like a direct connection to the network: the client is assigned an internal network address and can connect within the network just like an internal machine. It differs, though, in that the connection is made across the Internet itself, thereby offering the potential of faster and less expensive network connections. As implemented at the Laboratory, the VPN operates by taking over all communications to and from the client machine. Figure 2 shows a logical model of how various services (e-mail, WWW, etc.) establish separate connections between the client and the server. (Note that all of these connections run across the same physical wire, but that the software directs them to different "ports," which are software-controlled places to connect.)

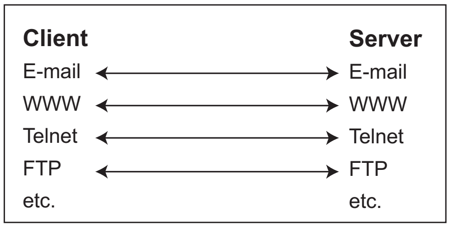


Figure 2. Logical Model of Normal Connections

Figure 3 shows how the VPN takes over all of these connections, forcing them through the shared VPN tunnel, with the connections being separated again at each end of the tunnel. If you are familiar with SSH (Secure Shell), you will recognize this as being similar to the way that SSH operates. The big difference is that SSH requires us to configure each tunneled connection separately, while the VPN automatically configures the tunneling without our needing to directly adjust the settings. With the VPN, we don't need to change Eudora's configuration; it works as is.

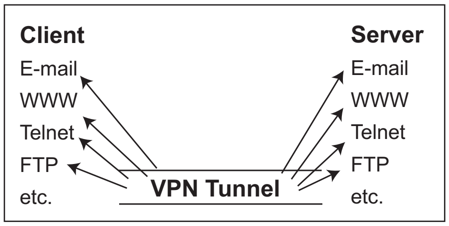


Figure 3. Logical Model of VPN Connections

Note that it is also possible to permit client connections outside of the VPN tunnel so that, for example, WWW connections would run through the tunnel to the Laboratory while e-mail connections could connect directly to an e-mail server on the remote network. Under our VPN security standard, however, no such outside connections are permitted for as long as the VPN connection is active. There are security risks to our network when a machine can connect to both it and an external network concurrently, so we don't allow it. (For further detail, see the Information Architecture Standard IA-9301: VPN Access to the Laboratory Unclassified Protected Network at http://www.lanl.gov/projects/ia/stds/ia930110.html.)

VPN Encryption

The VPN tunnel is secured from eavesdropping through encryption. As currently implemented at the Laboratory, we use a “DES within DES” approach. DES is the federal Data Encryption Standard (FIPS 46-2, and IA Standard IA-8307). DES was originally approved in 1977 and has served us well since then, but it is showing signs of age. The rapid increases in computer speed have made DES no longer as secure as it used to be, so we wrap DES within DES to make it more secure.

Before explaining this further, a couple of basics about DES and network communications may be helpful:

- DES works by encrypting data with a 56-bit “key” that is known by both the client and the server. One side encrypts the data with the key, and the other side decrypts using the same key. If this key is stolen or broken, then an eavesdropper can decrypt the data, which is the main weakness of DES. (A 1996 report found that a person with $400 to spend would need about 38 years to crack a 56-bit key, but that an organization with $300 million to spend could crack it in about 12 seconds. A 90-bit key, by contrast, would provide about 20 years of protection. See Footnote 1, below.)
- Network communications work by breaking data down into “packets”, 1.5 KB or smaller, which are reassembled by the receiving machine.

The way we currently implement VPN encryption at the Laboratory is illustrated in Figure 4. First, the entire VPN tunnel is encrypted in DES, so that everything inside of it is given a first level of encryption (using key “Y” in the illustration). Then, each packet within the tunnel is DES-encrypted with a separate, randomly generated key. In effect, this doubles the key length for each packet to 112 bits, and ensures that even if an eavesdropper were to crack one of the keys, only a single packet would be compromised—each packet needs to be cracked separately.

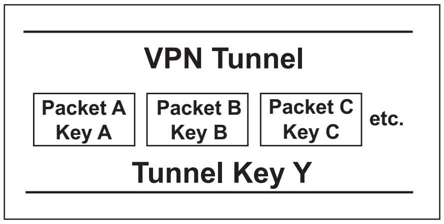


Figure 4. Basic VPN Encryption

One way of attacking encryption is to guess what the original text might be and to use that premise to crack the key. This is possible because most machine communications start with well known sequences. For example, when your Web browser requests a Web page, the request generally starts with “GET”.

To protect against this, we further strengthen the encryption through a technique called “Outer Cipher Block Chaining”, which injects random “spoiler data” into the data stream. With this approach, two identical blocks of text can be encrypted with identical keys, but the resulting “cipher text” will be different. To guess “GET”, the attacker would also have to guess the random spoiler data mixed into it.

Additional notes

- The initial connection is set up through “public key encryption,” which enables the client and server to connect without knowing each other’s secret key. Within this initial encryption, a DES key is negotiated between the client and server without the key itself ever being sent across the connection.
- Although the above description focuses on the use of DES, the Laboratory VPN can also be configured to use “Triple Pass DES” (112-bit key) or “3DES” (168-bit key). A “3DES within 3DES” set up would effectively have unique 336-bit keys for each packet, though the down side is reduced connection speed. Performance will be balanced against strength of encryption to determine which of the algorithms will be in use at a particular time, and the VPN server will be configured to automatically use that algorithm without the users needing to change their own software configuration.
- One measure of the strength of an encryption mechanism is the ability to publish it openly without increasing its vulnerability. All of the information in the above discussion is already widely published, and describing it here in BITS does not make the VPN connections more vulnerable. In other words, it’s okay to talk about how we do things, but it’s not okay to share, for example, the actual passwords that are used to log in to the VPN.

VPN Limitations

Before going into a discussion of the various ways that the VPN might be used, it may be helpful to note that there are limitations to what it does. The VPN gives us a lot of capabilities, but it doesn’t do everything.

The first thing to note is that the VPN does not encrypt the entire connection between the client and the destination server. Instead, as shown in Figure 5, the VPN only encrypts the connection between the VPN client and the VPN server.

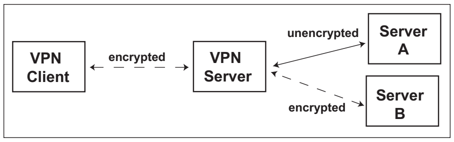


Figure 5. Limits to VPN Encryption

Connections from the VPN server to the destination server (such as a Web site being accessed through the VPN tunnel) may or may not be encrypted, depending on the destination server's configuration. If you go through the tunnel to access the main Laboratory home page, the connection between the VPN server and the Web server is not encrypted. But if you go through the tunnel to access the Web site for our Electronic Software Distribution (ESD), the entire connection will be encrypted. (In this particular case, because ESD is configured to use SSL/DES, the encryption within the tunnel itself would effectively be "DES within DES within DES.")

If you are operating a server which has data that requires encrypted transfer within the Laboratory Unclassified Protected (blue) Network, the VPN alone does not provide that. Your server still needs to be configured to support encryption.

A login to the central VPN service only establishes that an individual is authorized to access the Laboratory Unclassified Protected Network. It does not establish any permissions beyond what any other user of the protected network has. For example, it does establish permission to access Web pages that are restricted to Laboratory machine addresses only, but it does not in itself establish permission to access protected user accounts within the network. (Unlike Kerberos, it does not support forwardable tickets, though Kerberos can operate through the VPN connection.)

As of this writing, the VPN itself has not been approved for the transfer of non-Laboratory owned information such as UCNI (Unclassified Controlled Nuclear Information). If, however, connections have been otherwise approved for transferring such information, then they can be run through the VPN tunnel.

A final limitation is that when your machine is connected to the VPN, all communications from your machine will go through that connection. This means that all of your communications will go through the Laboratory network and will be subject to official use policies. If, for example, you have connected from your own machine and would like to check your bank account (personal use), you should disconnect from the VPN before doing that.

VPN Applications

Within the above limitations, there are still a number of ways we can use VPN software to accomplish our work. The most common has already been outlined above, where a remote user uses the VPN to connect to the Laboratory network and perform tasks as if he/she were a local user. The security model calls for an exclusive connection from the client in this case, and no other machines are allowed access through the connection.

The VPN can also be used to connect an entire remote network (subnet) to the Laboratory network, so that all users of that network have shared access to the Laboratory. A pocket of Laboratory users in, for example, Nevada could use the VPN to tunnel through others' networks and use the Laboratory network. In this case, the security model calls for all communications from the remote network to pass through the tunnel (i.e., no other Internet connections) and for all users of the remote network to be authorized users of the Laboratory Unclassified Protected (blue) Network (i.e., no unauthorized guests).

For users who would like additional security over modem connections, the VPN can be run across dial-in modem lines. If you have connected to the Laboratory's central dial-in modem service, for example, the VPN can be run across that connection, although that's probably overkill in most cases. (If you've connected to the central dial-in modems, you've already established authenticated access to the Laboratory network, and the only thing the VPN adds is encryption between the remote machine and the VPN server.)

If you have dialed in to an external ISP (Internet Service Provider), the VPN can be run across that connection, too. This can be useful if, for example, you're having troubles connecting to the central Laboratory dial-ins. Authorized users in Virginia could connect to a Virginia ISP via local phone lines and use the VPN to reach the Laboratory via the Internet (as opposed to using the long-distance lines to the Laboratory dial-ins).

How to Get the Software

The Laboratory central VPN service is currently based on the Shiva VPN Suite by Shiva Corporation, which recently became a subsidiary of Intel. Authorized Laboratory users can purchase the preconfigured Shiva VPN Client at the Laboratory's Electronic Software Distribution Web site (ESD, http://esd.lanl.gov/). Others can
get the client through Shiva at http://www.shiva.com, though they will need to configure it themselves. As of this writing (March 1999), the Shiva VPN Client is only available for Windows 95/NT. A Macintosh version is expected this summer.

The current solution is a proprietary solution, so other VPN clients do not yet work with it. VPN standards are emerging, however, such as the IETF (Internet Engineering Task Force) Internet Draft for the Point-to-Point Tunneling Protocol (PPTP). As such standards become more stable, we will hopefully see a wider variety of interoperable software.

For Additional Information

For additional information about the Laboratory VPN, including links to configuration information, please see IA-9301: VPN Access to the Laboratory Unclassified Protected Network at http://www.lanl.gov/projects/ia/stds/ia930110.html. For a good, more general discussion of the Shiva VPN, please see “Shiva Virtual Private Networking Concepts” at http://www.shiva.com/pdf/shivavpnconcepts.pdf. For additional information about the Information Architecture Project in general, please see our project home page at http://www.lanl.gov/projects/ia/.

Footnote 1: Blaze, et al., “Minimal Key Lengths for Symmetric Ciphers to Provide Adequate Commercial Security,” 1996.http://www.counterpane.com/keylength.html.

ASCI Training Course Descriptions

Fee: No charge

Length of each class:
Approximately 2 hours

Registration: Contact Leslie Linke (505-667-9399 — lal@lanl.gov) OR Susan Simonsic (505-667-9559 — sues@lanl.gov).

Schedule: Classes will be scheduled as soon as there is sufficient enrollment

Getting Started on ASCI Blue Mountain

Presenter: Joe Kleczka

This talk is an introduction to ASCI Blue Mountain and production computing using the IRIX Origin 2000 cluster machines, where to get information, and how to get started. Hardware and software configuration will also be discussed.

Running Message-Passing Interface (MPI) on Blue Mountain

Presenter: MaryDell Tholburn

This class covers resources used by the SGI/MPI library (array services and HIPPI), MPI buffers, MPI environment variables and what they do, MPI modules and .files, MPI statistics, launching an MPI job, and troubleshooting.

TotalView

Presenter: Lauren McGavran

A basic TotalView tutorial is presented, covering what you need to know to debug a single process and an MPI code. Topics covered include how to get around in TotalView, examine variables, set various kinds of breakpoints, and other basic debugging techniques. An overview of the more complex features of this debugger, with details as requested, will be provided.

Load Sharing Facility (LSF)

Presenter: Stephany Bouchier

The LSF class explains why the LSF software is used for job control on the Origin 2000 systems at LANL and describes how the systems are configured. The major emphasis of the class is to teach users how to submit jobs to run on the system, query LSF about its configuration, or query LSF about jobs which have been submitted.

High Performance Storage System (HPSS)

Presenter: Mark Roschke

This class provides an introduction to the HPSS. Various aspects of the Parallel Storage Interface (PSI) will be covered, including transferring files, listing and changing file attributes, and interface customization.

Library Databases

A Few of the New Databases Available at the Research Library

Search the Social Sciences!: *Social SciSearch® at LANL*

by Jeane Strub,
Databases Team Member
CIC-14, Research Library

Need access to economics, industrial relations, law, political science or other social science literature? All this is now available electronically from then Research Library through the Social SciSearch® at LANL database. Corresponding to the print Social Sciences Citation Index, it is produced by the Institute for Scientific Information and is a sister database of SciSearch® at LANL.

Updated weekly, Social SciSearch® at LANL covers 1,700 worldwide journals in a broad range of disciplines and includes all significant items such as articles, letters, editorials, etc. It also covers individually selected, relevant items with a social science focus from over 3,300 of the world's leading scientific and technical journals. In addition to searching by traditional methods such as author, title, and abstract words, cited reference searching is also available. A known paper is searched to identify more recent papers that cite the earlier work. In this manner other papers can be identified which might be of interest.

Hyperlinks are provided where the article or journal is available full-text on the World Wide Web and holdings in the LANL Research Library are noted in the full record. The same Weekly Alerting Service available for SciSearch® at LANL is also available in Social SciSearch® at LANL. Users may register for this service and then create a search strategy that will be matched automatically against each new weekly update.

This database is accessible at http://scisearch2.lanl.gov:8082/lanl/ or by selecting Electronic Databases from the Research Library home page and then selecting Social SciSearch. If you have any questions about using this product please call the reference desk at 7-5809. Check the training schedule for upcoming free classes on the database.

CNS Nonproliferation Databases available Lab-wide

by Irma Holtkamp,
Databases Team Member
CIC-14, Research Library

The Center for Nonproliferation Studies (CNS) of the Monterey Institute for International Studies maintains five valuable databases that are now available to Laboratory researchers through the Research Library. Until recently access to these databases, available only by subscription to the CNSData service, was limited to selected staff within LANL. The Research Library in collaboration with CISA, NIS-DO, NIS-7, NIS-8, NIS-9 and NAC has arranged to make these databases available throughout the Laboratory to all interested researchers.

The CNS databases contain continually updated information compiled from over 340 source publications, including United Nations and International Atomic Energy Agency documents, trade journals, government and defense publications, periodicals and electronic news sources, academic journals, US congressional testimony, conference proceedings, book chapters, correspondence from international advisors, unpublished papers, and Internet sources. Not only bibliographic references, but often full-text documents are provided. Subject coverage is the global proliferation of weapons of mass destruction (nuclear, chemical, and biological) and their delivery systems.

To use these databases, onsite LANL researchers can

- connect to http://cns.miis.edu/db/index.htm directly, or
- select "CNS databases" from the Library's Web site map, or
- select from the Library's Electronic Databases pages by name (use the Alphabetic listing or search on keywords like "CNS") or by Subject category, "International Affairs."

When offsite, LANL researchers can use a login/password available from the Library Service Desk (e-mail: library@lanl.gov or phone: 7-5809). Please send questions and comments regarding these databases to: db-info@lanl.gov.

Library Without Walls Selected by Elsevier Science for Advanced Technology Partnership

by Rick Luce, LWW Project Leader
CIC-14, Research Library
Frances Knudson, LWW Team Member
CIC-14, Research Library

Capitalizing on the Research Library's success providing access to electronic journals, Elsevier Science has selected the Library Without Walls (LWW) program to be an Advanced Technology Partner. Under terms of the agreement, LANL receives all of the approximately 1,200 Elsevier Science electronic journal titles, beginning with 1995 publications. The agreement brings 684 new journal titles to the Lab research community, at an estimated LANL savings of over $917,000. Titles include *Journal of supercritical fluids, Computers & electrical engineering, International journal of plasticity, Space technology,* and *Water policy.*

Access to the Elsevier titles for LANL researchers is provided through article-level links in the Research library's locally loaded databases such as SciSearch® at LANL and INSPEC® at LANL (available on the Electronic Databases page, http://lib-www.lanl.gov/edata/edata.htm). The Elsevier titles are also available through a journal browse application. ScienceServer LLC has been selected to provide the journal server application, which manages roughly 700 GB of Elsevier data. To use the journal browser, click on a journal title appearing on the listing of Elsevier journals (http://lib-www.lanl.gov/ejournals/PUB_Elsevier.htm). Elsevier journals are also included on the Alphabetical and Subject listings on the Electronic Journals page (http://lib-www.lanl.gov/cgi-bin/ejrnlsrch.cgi).

Under the partnership the LWW will develop and support the capability to provide licensed access to the Elsevier titles to other members of the New Mexico Library Alliance, beginning with the Air Force Research Laboratory.

Editor's Note: For a listing of other new databases see this Web site: http://lib-www.lanl.gov/edata/edata.htm.

Parting Comments from a Retiring Computing Veteran

by Denise Sessions,
BITS Managing Editor
CIC-6, Customer Service

Advanced computing capabilities at the Laboratory are positioned to capitalize in the public sector thanks to top staff, leadership support, and innovative technology, predicted a retiring 23-year computing veteran.

Ann Hayes, Executive Director of the Advanced Computing Laboratory (ACL), retired on April 2. BITS interviewed Ann before she left for her perspective on the future of computing at Los Alamos. In the division since 1976, Ann recalled making the transition from a technical position to management. She reminisced about fostering computational science and simulation to "come into its own", planning the potential move of the CM-5 computer from the ACL to the Laboratory Data Communications Center (LDCC), and surviving two floods in the ACL. Observing the transitions in her career, she remembered a slowdown in innovation in the mid '80's, but a wild ride since then.

From TSM to Group Leader to Executive Director

"As I leave here, the most fun part of the whole thing has been working with people," Ann said. "There's a lot of really good people coming up." About five years into her management position, Ann discovered a different kind of job satisfaction being a manager versus a technical staff member. Although at first management didn't seem as satisfying as technical work, she discovered the rewards to be greater as time went on. For example, Ann found it very rewarding to watch people grow in their careers.

I'm probably the only person who would tell you this, but I foresee a time when we have truly distance computing, distributed across the country.

Ann was the C-3 group leader from 1980 before joining the ACL in 1989 as one of its original six employees. The ACL evolved from colleague Andy White's dream of research collaborations with smaller, innovative computers. White envisioned matching unique machines like the CM-2 with researchers who wanted to learn to program those machines.

During Ann's tenure at ACL, CM-5 was threatened twice by flooding. The first time, Ann received a phone call about 9 p.m. from a late-working staff member. When she arrived at the facility, she and the staff member covered the CM-5 and everything else they could until the Fire Department arrived. Ceiling tiles fell like rain. A year later, it happened again. Now a custom-made, red cover over the computer makes it look like an elephant. However, its blinking lights continue to demonstrate innovative programming at work at Los Alamos.

From Innovative Hardware to Software Frameworks

Ann highlighted some of ACL's accomplishments. Initially they put Thinking Machines™ on the map and did innovative programming. They started the first Thinking Machines™ user group. When it became apparent that niche marketing wasn't going to survive, ACL decided to go further with runtime software that could run on different machines rather than pursue unique hardware. When Accelerated Strategic Computing Initiative (ASCI) codes got started, researchers at ACL used the Parallel Object-Oriented Methods and Applications (POOMA) framework and the suite of software it spawned as part of the runtime-software effort. The researchers started using POOMA to simplify the development of scientific-application codes across rapidly evolving high-performance computing architectures.

ACL has contributed to making the Laboratory competitive in the larger world of computing. This competitiveness is possible because ACL was instrumental in helping the division set its sights externally as well as internally in terms of peer recognition. In addition, ACL was instrumental in getting the Laboratory to recognize the value of industry collaborations. An important accomplishment is that DOE/Energy Research has ranked ACL among its flagships, especially due to its work on the Grand Challenge problems and usable parallel hardware.

Ann noted that Los Alamos has always benefited from its reputation for good science. "That's the best part of all the jobs," she said. Also, the Internet has contributed to the Laboratory's future collaboration, communication, and the ability to get things done. It's hard now to imagine when the Internet wasn't there. I remember the first days, when you could get on and talk (laboriously)," Ann recalled. "It was so exciting!" She started out on the Silent 700, the first PC. Of course, before that she used punch cards and paper tape!

For the first time I can remember, funding at LANL is great for computing.

For the Future: Make Good on the Commitments

Considering these accomplishments, Ann made some recommendations about the future. Her advice is for us be aware of who funds what. She added that the Lab needs to deliver on commitments to different funding agencies and give credit where credit is due. We should tout accomplishments in terms of the sponsors. Ann explained how to make good on the commitments**.** "We have more public customer buy-in because most simulations we do for the weapons side have counterparts in society." Examples are the wildfire simulation and global-modeling capabilities, which came from weapons hydrodynamics and turbulence codes. Also, she added, the technology is almost there.

She thinks we have the right combination of good scientists, software developers, computer hardware, and software to bring it all together. The support of the Lab's director, as evidenced in making computational modeling one of the Lab's core competencies, is a key component. Also, there's been a breakthrough in being able to offer competitive salaries to attract top candidates to be competitive with industry. The ACL has not only come up with the money to offer competitive salaries, but ACL staff take time to introduce candidates to everyone at the facility when prospective job candidates interview. ACL staff also give candidates and their families opportunities to the look at local housing and schools.

Distance-Distributed Computing

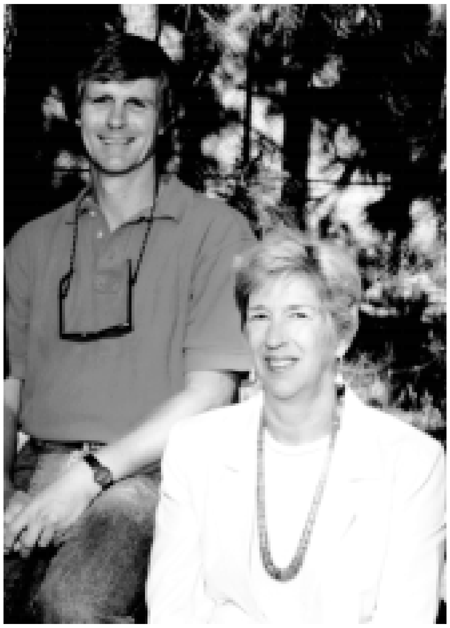
Because of fast networking and storage, it is now possible to push to make smaller parallel computers so that they take up less-expensive real estate. Ann's vision is in distance distributed computing. "I'm probably the only person who would tell you this, but I foresee a time when we have truly distance computing, distributed across the country. You will have your desktop computer, connected to powerful supercomputers, but all that 'big iron' will remain at the manufacturer's site, where their own support people are available to work on it." The idea is similar to the ability we now have to access a library's card catalog without going to the library. She got this idea while talking about the costly maneuver of moving the CM-5 from ACL to the LDCC to make it a secure machine. It turned out that it was less risky to create a vault around the CM-5, making part of the ACL a secure facility, than to dismantle and reassemble a machine whose parent company had gone out of business.

No Limit on Computing Power

As for the future of computing speed, Ann predicted that in the next few years, there will be a continuing use of lots of processors and improving connections between machines. Since 1992, researchers in computing talked about limits on the size of chips because of heat dissipation. Then came massively parallel computing. "There is no limit on power; massively parallel systems will keep going," Ann commented. "Look at the things people wouldn't have thought possible 10 years ago that are now commonplace. The power of a room full of computers is now in something like a Palm Pilot. Optical computers are coming. Better connections between processors will make the next increments in computing speed possible."

We have wonderful people, exciting work, and a high level of support from the Lab as a whole.

Ann is optimistic about the future of the incredibly fast-changing scientific discipline she's been involved with for the last 23 years. "For the first time I can remember, funding at LANL is great for computing." She believes we now have great computing opportunities. "We are lucky to be in this Lab in this division at this time," she commented. "We have wonderful people, exciting work, and a high level of support from the Lab as a whole."



Programming and Application Training Summer Course Offerings

Attention Summer Students

These programming and application training courses are available this summer. Any employee may register but summer students are encouraged to take advantage of this training opportunity. Sign up as soon as possible because the enrollment is first come, first served. Also, if there is not sufficient enrollment, a course may be cancelled. To register go to the Laboratory's Virtual Training Center Web site and click on Computer: http://www.lanl.gov/internal/training/training.html; or call CIC-6 Training, Development, and Coordination Team at (505) 665-4444, option 4; or send e-mail to cic6-train@lanl.gov.

Java Programming
June 21–25, 1999,
8:30 a.m. – 5:00 p.m.

SM-200 Classroom [TA-3, SM-200, Room 115 (an unclassified area)]

Topics: Using Java to Create Applications and Applets; Defining and Describing Garbage Collection, Security, and the Java Virtual Machine; Describing and Using the Object-Oriented Features of Java; Developing Graphical User Interfaces; Taking Advantage of the Various Layout Managers Supported by Java; Describing and Using the Java 1.1 Delegation Event Model Using Java Windowing Components, Including Mouse Input, Text, Window, and Menu Components; Using Java Exceptions to Control Program Execution and Define Custom Exceptions; Using the Advanced Object-Oriented Features, Including Method Overriding and Overloading, Abstract Classes, Interfaces, Final and Static, and Member and Field Access Control; Perform File Input/Output; Built-In Threading Models to Control the Behavior of MultipleThreads; and Accessing Servers and Clients Through Sockets.

Beginning Unix
July 6–9, 1999,
8:30 a.m. – 12:00 noon

SM-200 Classroom [TA-3, SM-200, Room 115 (an unclassified area)]

Description: This basic course discusses capabilities available on the UNIX operating system.

Topics: Overview of the Workstation Environment; Getting Started; the UNIX File System; Manipulating Files; Customizing Your Environment; the C-Shell; Editing and Writing with vi; Using the Network; Discussing NFS and NIS; Using Basic System Status Commands; Startup and Shutdown; Procedures; and Using tar.

Advanced Unix
August 10–13, 1999,
8:30 a.m. – 12:00 noon

SM-200 Classroom [TA-3, SM-200, Room 115 (an unclassified area)]

Prerequisites: Completion of Beginning UNIX class or equivalent knowledge.

Topics: File Manipulation; File Reorganization; Network File System Concepts; Introduction to C-Shell Scripts; Conditional Execution; Shell Programming; The Korn Shell; Korn Shell Script Features; and SED Filtering Tool.

Windows NT Workstation and Server
August 2–6, 1999, 8:30 a.m. – 5:00 p.m.

Los Alamos Inn

Description: This course is valuable for employees who are evaluating whether to migrate or have already migrated to Windows NT. It benefits system and network administrators, other support personnel, programmers, and users from Windows, UNIX, OS/2, or VMS backgrounds.

Topics: Introduction to Windows NT; System Overview and Security; Network Configuration Options; Installation; Server Choices; User Administration and Security; Files and Printers; Built-In Network Support; Configuration Options; Using Setup; Data and Disk Management; The Registry; Troubleshooting; and Optimization and Performance.

Research Library Training Schedule

The LANL Research Library offers a variety of training opportunities for the Laboratory community. Sessions focus on specialized library databases and other electronic resources. While the sessions listed below will be held at the library, training can be arranged at your site. Contact the Library by phone at 7-4175 or by e-mail to library@lanl.gov, to register for a session or to arrange a special session/training at your site.

Date	Time	Session Title
5/20/99	2:00-4:00	Infosurfing: Basic Web Searching Strategies
5/26/99	1:00-1:30	Research Library Tour
6/2/99	1:00-1:30	Research Library Tour
6/2/99	1:30-2:00	Introduction to Electronic Library Resources
6/8/99	1:00-1:30	Grants & Funding on the WWW
6/9/99	1:00-1:30	Research Library Tour
6/10/99	1:00-1:30	SciSearch“/Social SciSearch“ at LANL
6/15/99	1:00-1:30	What the Report Collection Can Do for You
6/16/99	1:00-1:30	Research Library Tour
6/23/99	1:00-1:30	Research Library Tour
6/23/99	1:30-2:00	Introduction to Electronic Library Resources
6/24/99	2:00-4:00	Infosurfing: Basic Web Searching Strategies
6/29/99	1:00-1:30	Earth & Environmental Resources on the WWW
6/30/99	1:00-1:30	Research Library Tour
7/1/99	1:00-1:30	Alerting Service for the SciSearch“& Social SciSearch“Databases
7/7/99	1:00-1:30	Research Library Tour
7/14/99	1:00-1:30	Research Library Tour
7/14/99	1:30-2:00	Introduction to Electronic Library Resources
7/15/99	1:00-1:30	GeoRef“ on the Web
7/20/99	1:00-1:30	Melvyl“ (U of CA specialized databases)
7/21/99	1:00-1:30	Research Library Tour
7/22/99	2:00-4:00	Infosurfing: Basic Web Searching Strategies
7/28/99	1:00-1:30	Research Library Tour
8/4/99	1:00-1:30	Research Library Tour
8/4/99	1:30-2:00	Introduction to Electronic Library Resources
8/5/99	1:00-1:30	Grants & Funding on the WWW
8/10/99	1:00-1:30	BIOSIS“ on the Web
8/11/99	1:00-1:30	Research Library Tour
8/18/99	1:00-1:30	Research Library Tour
8/26/99	2:00-4:00	Infosurfing: Basic Web Searching Strategies

Technical and Advanced Technical Computer Training Courses

Communications	**Office Skills 2000**	**Web Authoring and Browsing**
• Eudora 4.x • Meeting Maker 5.0.3	• Office Skills 2000—LANL • Office Skills 2000—Professional Development	• Dreamweaver 2.0–MAC or PC • FrontPage 98 • HTML Basics • HTML Intermediate
Enterprise Information Applications (EIA)	**Other EIA Courses**	**System Administration Training**
• Date Warehouse–Basics • Date Warehouse–EDS Reports • EDS—Basics • EDS–GUI • EDS–Training Plans • Infomaker • Invoice Approval System • Purchase Card System • Procurement Desktop • Recharge • Time & Effort GUI • Travel Foreign GUI • Travel Domestic GUI • Web JIT	• Financial Management Information System (FMIS) • Property Accounting, Inventory and Reporting System (PAIRS) • Signature Authority System (SAS) • Secretarial/Contract Service (SE) • Salary Review System (SRS) • Directory Information System (DIS) • Automated Chemical Information System (ACIS)	• SGI System Administration (Beginning) • SGI System Administration (Advanced) • SGI Network Administration • SGI Performance Evaluation and System Tuning • Solaris 7 System Administration • Solaris 7 Network Administration • Solaris 7 Server Administration • Unix and Widows NT Integration • Windows NT Workstation and Server • Windows NT Optimization and Troubleshooting • Windows NT Security
Programming Training	**Application Training**	**ASCI**
• C Programming (Beginning) • C Programming (Advanced) • C++ for Experience C Programmers • ANSI/ISO C++ Programming Clinic (Advanced C++) • Distributed Objects Using Corba • Java Programming • Java Program Workshop • Distributed Programming with Java • Object Technology–A Management Overview • Object-Oriented Analysis and Design • Perl Programming • Advanced Perl Programming with CGI • C-Shell Programming • Programming for Beginning using Java	• Advanced WWW Development • FrameMaker Basic and Advanced • Foundations of IDL Programming • IDL 5.0 Graphic Object Workshop • Netscape Servers for Intranet Development • Origin 2000 Applications Programming and Optimization • Sendmail–Managing Internet Mail • C++ and the Unified Modeling Language • Sybase Fast Track to Adaptive Server Enterprise 11.5 (ASE) • Sybase Performance and Tuning for System 11 • Sybase SQL Server Administration • Unix (Beginning) • Unix (Advanced) • Visual Basic 5.0 Fundamentals • Visual C++ Windows Programming	• Mastering Projects Workshop • Software Engineering for Scientists and Engineers • Getting started on ASCI Blue Mountain Systems • Running MPI on Blue Mountain Systems • Introduction to Totalview • LSF (Load Sharing Facility) • Introduction to HPSS (High Performance Storage System)

** You do not need an ICN password to use e-mail.

1999 12-Month Index

For a more complete index, see <http://www.lanl.gov/orgs/cic/bits/index/index_home.html

Keywords	Title of BITS Article	Date	Page
ASCI	*Blue Mountain IS World's Fastest Computer*	*Dec. 98*	*1*
BITS Announcements	*BITS Gets New Design, New Editor*	*F/M 99*	*3*
BITS Interviews	*Rick Luce*	*Apr. 98*	*1*
		J/J 98	*5*
	Gary Clark	*J/J 98*	*2*
	John Blaylock	*A/S 98*	*2*
	Chris Kemper	*A/S 98*	*8*
	Bob Newell	*O/N 98*	*1*
	Earleen Eden	*O/N 98*	*3*
	Dale Land	*Dec. 98*	*11*
	John Morrison	*F/M 99*	*6*
CIC (Computing, Information & Communications)	*BITS Welcomes New Deputy Division Director for Strategic Computing*	*F/M 99*	*6*
CIC-2	*CIC-2's Future Is in Stewardship, Automation, and "Fun Stuff"*	*Dec. 98*	*11*
CIC-10	*CIC-10 "Knowledge Is Our Most Important Product"*	*O/N 98*	*3*
CIC-12	*CIC-12 Moves Into Sofware Quality Assurance, Needs Scientific Programmers*	*J/J 98*	*2*
CIC-14	*CIC-14 Becomes a Bridge to Human Knowledge*	*Apr. 98*	*1*
	Library Without Walls Anticipates Its Next Generation	*J/J 98*	*15*
Desktop Computing	*CIC-2's Future Is in Stewardship, Automation, and "Fun Stuff"*	*Dec. 98*	*11*
Desktop Computing Costs	*Eye-Opening Reports About Desktop Computing Costs*	*J/J 98*	*10*
E-mail	*MacTips: Dealing with [E-mail] Attachments in Eudora Pro*	*Aug. 97*	*16*
Embedded Systems	*The Year 2000 Bug is Hiding*	*F/M 99*	*1*
ESD	*Working for and Serving You*	*F/M 99*	*10*
Filters	*Content Labeling and Signing: Getting Ready for Filters*	*Apr. 98*	*10*

Keywords	Title of BITS Article	Date	Page
Fraud	*Detecting Tax and Medicare Fraud*	*J/J 98*	*4*
HCPwire	*Lab Considers Group Subscription for HCPwire*	*Apr. 98*	*4*
Information Systems	*Engineering Index Is Now Available from the Research Library*	*Dec. 98*	*13*
	Research Library Adds Social SciSearch® Database	*F/M 99*	*5*
JavaScript	*JavaScript Observations and Tips: Part II*	*May 97*	*11*
Labwide Systems	*Customer Feedback Guides Improvements to Labwide Systems*	*Aug. 97*	*4*
Newell, Bob	*24 Years Later, Bob Newell Joins the Lab as Deputy CIC Division Director*	*O/N 98*	*1*
Networks	*Laboratory Unclassified Network Will Implement Changes for Secuirty*	*Dec. 98*	*3*
	Preserving Bandwidth on Your Windows NT Network	*F/M 99*	*8*
PNG	*Its' Time for PNG: A Graphics Format You Can Pronounce*	*O/N 98*	*7*
Popular Science Award	*PS Awards LANL Web*	*O/N 98*	*12*
Radiation Transport	*Unstructed-Mesh Radiation Transport*	*J/J 98*	*1*
RAGE	*The Los Alamos RAGE Code: Scalable, Parallel Performance*	*Apr. 98*	*5*
Records Management	*CIC-10: "Knowledge Is Our Most Important Product"*	*O/N 98*	*3*
SciSearch	*Research Library Adds Social SciSearch® Database*	*F/M 99*	*5*
Supercomputing	*Blue Mountain Is World's Fastest Computer*	*Dec. 98*	*1*
Wildfires	*Forecasting Wildfires and other Crises*	*Apr. 98*	*3*
Windows NT	*Preserving Bandwith on Your Windows NT Network*	*F/M 99*	*8*
World Wide Web	*Research Library's WWW Online Catalog Improved*	*Nov. 97*	*10*
(WW or Web)	*Tips for Desktop Users Available on the Web*	*Apr. 98*	*8*
	Enterprise Information Applications Available on the Web	*J/J 98*	*8*
	Research Library Now Offering INSPEC Database	*J/J 98*	*9*
	It's Time for PNC: A Graphics Format You Can Pronounce	*O/N 98*	*7*
	Lab Web Publishers Invited to IntraLab99	*F/M 99*	*7*
Year 2000 (Y2K)	*The Year 200 Bug is Hiding*	*F/M 99*	*1*

Customer Support Center . . . . (505) 665-4444, ext. 851, or cichelp@lanl.gov

Because of a wide variety of CIC computing services, numerous facilities are available to address your questions. If you are uncertain whom to call, you can always call the Customer Support Center (CSC). CSC consultants are trained to either answer your question or locate someone who can. To reach the appropriate consultant, dial 665-4444 and make your selection from the following choices:

Option 1: New user topics including e-mail, passwords, registration, and World Wide Web
Option 2: Enterprise Information Applications such as Travel, Time and Effort, and Purchase cards
Option 3: Scientific computing, storage systems, and networking
Option 4: Classroom instruction and training
Option 5: Desktop Consulting for PC and Macintosh software and network configurations.

Consulting Via E-Mail

Customer Support	cichelp@lanl.gov
Scientific and engineering computing........	consult@lanl.gov
Administrative and business computing	eiaconsult@lanl.gov
Passwords and registration	validate@lanl.gov
Macintosh computing	Mac-help@lanl.gov
PC computing	PC-help@lanl.gov
UNIX computing	UNIX-help@lanl.gov

Other Useful Numbers

Advanced Computing Laboratory	665-4530
Central Computing Facility	667-4584
Network Operations Center	noc@lanl.gov or 667-7423
Telephone Services Center	667-3400

Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36.

All company names, logos, and products mentioned herein are trademarks of their respective companies. Reference to any specific company or product is not to be construed as an endorsement of said company or product by The Regents of the University of California, the United States Government, The U.S. Department of Energy, nor any of their employees. The Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; therefore, the Laboratory as an institution does not endorse the viewpoint of a publication or guarantee its technical correctness.

Los Alamos
NATIONAL LABORATORY

Los Alamos
NATIONAL LABORATORY

BITS is published bimonthly to highlight recent computing and communications activities within the Laboratory. We welcome your suggestions and contributions.

BITS may be accessed electronically at this URL:
http://www.lanl.gov/cic/publications.html

LALP-99-42 (6–7/99)